PATENT
Howrey Docket No. 01827.0044.00US00
00CXT0363

SYSTEM FOR CARRIER PHASE TRACKING OF MULTI-DIMENSIONAL CODED SYMBOLS

RELATED APPLICATIONS.

This application is related to "RATE N/N SYSTEMATIC, RECURSIVE CONVOLUTIONAL ENCODER AND CORRESPONDING DECODER," U.S. Patent Application Serial No. 09/602,690, Howrey Dkt. No. 01827.0037.00US00, Conexant Dkt. No. 00CXT0357D, filed June 23, 2000; "A SLIDING WINDOW TECHNIQUE FOR MAP DECODERS," U.S. Patent Application Serial No. 09/629,122, Howrey Dkt. 01827.0041.00US00, Conexant Dkt. No. 00CXT0360D, filed July 31, 2000; "SYSTEM FOR CARRIER PHASE TRACKING OF CODED SYMBOLS USING RELIABILITY METRICS FOR SYMBOL ESTIMATES", U.S. Patent Application Serial No. 09/715,877, Howrey Dkt. No. 01827.0042.00US00, Conexant Dkt. No. 00CXT0361D, filed on November 17, 2000; and "ITERATIVE CARRIER PHASE TRACKING SYSTEM," U.S. Patent Application Serial No. Not Yet Assigned, Howrey Dkt. No. 01827.0043.00US00, Conexant Dkt. No. 00CXT0362D, filed on December 4, 2000. Each of these applications is owned in common by the assignee hereof, and each is hereby fully incorporated by reference herein as though set forth in full.

BACKGROUND OF THE INVENTION

1. Field of the Invention.

This invention generally relates to multi-dimensional, coded symbols, and, more specifically, to carrier phase tracking of such symbols.

2. Related Art.

The carrier phase of a signal can meander with time, due to instabilities in the transmitter upconversion circuitry, or instabilities in the demodulator oscillator and downconversion circuitry. The presence of this phase noise degrades the performance of receiver, by creating a phase rotation in the actual signaling constellation with respect to the assumed signaling constellation. Since the phase noise typically varies at a much slower rate than the transmitted symbol rate, this phase noise trend can often be estimated (e.g. "tracked"), and subsequently compensated for by circuits within the receiver.

Conventional tracking loops typically operate on one symbol at a time. A symbol is compared to an estimate of the symbol, and a phase adjustment made to the next symbol in response to this comparison. The frequency at which the loop is updated is typically the frequency at which new symbols are introduced to the tracking loop.

A multi-dimensional symbol is essentially a plurality of symbols that bear a logical relationship to one another in that all are encoded from the same group of underlying source bits. Conventional tracking loops process multi-dimensional symbols by processing each of the individual symbols in the plurality one at a time. This approach is not optimal because it ignores the fact that the symbols bear a logical relationship to one another. It is also computationally burdensome because some of

the analysis performed on each symbol is redundant and unnecessarily repeated from symbol to symbol.

SUMMARY

The invention provides a system for carrier phase tracking of multi-dimensional coded symbols. A multi-dimensional symbol is input to a symbol estimation module. The multi-dimensional symbol can be represented by the vector r_k^D, where the subscript k indicates time, and the superscript D indicates that the vector comprises a plurality D of individual constituent symbols. The symbol estimation module provides an estimate for the multi-dimensional channel symbol. The estimate itself may be multi-dimensional, and represented by the vector s_k^D. This indicates that estimates of the individual constituent symbols are included within or easily derivable from s_k^D.

The multi-dimensional symbol estimate s_k^D may be formed from any device capable of producing soft symbol estimates, including, without limitation, a MAP decoder, a log-MAP decoder, a SISO decoder, a SOVA decoder, a Viterbi decoder, or the like. In one implementation, a modified log-MAP decoder is used where only the alpha (forward recursion) engines are utilized.

A residual determination module forms a multi-dimensional residual indicating a difference between the symbol r_k^D and the estimate s_k^D. The residual may be a scalar z_k and lack multiple dimensions, or it may be a vector z_k^D, and have multiple dimensions, each representing a residual between an individual constituent symbol, and the estimate of that individual symbol. The residuals may be, without limitation, phase residuals or they may be orthogonal component residuals, i.e., the component of a symbol orthogonal to the estimate of that symbol.

A scalar reliability metric R_k, indicating the reliability of the multi-dimensional estimate, may also be provided by the symbol estimation module. The reliability metric may be used to weight the corresponding residuals. If the residual is the vector z_k^D, the weighted residual may be represented by $R_k \bullet z_k^D$, indicating that

R_k is used to weight each of the components of z_k^D. If the residual is the scalar z_k, the weighted residual may be represented by $R_k \bullet z_k$.

In one application, the tracking loop module then determines, responsive to one or more of the weighted or unweighted residuals, a derotation phase θ_k for the multi-dimensional symbol. This derotation phase may be a scalar θ_k, which applies to each of the individual symbols in the multi-dimensional symbol. Or it may be a multi-dimensional vector θ_k^D, including individual derotation phases for each of the individual constituent symbols in the multi-dimensional symbol. In another application, the tracking loop module then determines, responsive to the weighted or unweighted residuals, a phase offset estimate for the multi-dimensional symbol. The phase offset estimate may be a scalar $\Delta\theta_k$, which applies to each of the individual symbols of the multi-dimensional symbol r_k^D. Or it may be a vector $\Delta\theta_k^D$, with individual phase offset estimates for each of the individual constituent symbols in the multi-dimensional symbol. These derotation phases or phase offset estimates may be used to derotate the individual symbols in the multi-dimensional symbol to reduce or eliminate phase offset with respect to the carrier.

In one example, the residual which is produced is the vector z_k^D, and the same reliability metric R_k is used to weight each of the components of the vector z_k^D. The tracking loop module in this example determines an individual component of θ_k^D or $\Delta\theta_k^D$, as the case may be, responsive to the corresponding individual weighted residual. In other words, it determines θ_k^i or $\Delta\theta_k^i$, $1 \leq i \leq D$, responsive to $R_k \bullet z_k^i$, $1 \leq i \leq D$.

In a second example, a function of the individual component residuals of z_k^D is formed. Such a function may be designated as $f(z_k^D)$. For example, $f(z_k^D)$ may be the average of the individual component residuals. This function may be weighted by the reliability metric for the multi-dimensional channel symbol. The tracking loop module may then determine a scalar derotation phase θ_k or phase offset estimate $\Delta\theta_k$

responsive to the value $R_k \bullet f(z_k^D)$. This value may then be applied to each of the individual symbols in the multi-dimensional symbol

In a third example, a composite residual is formed for a multi-dimensional symbol. The composite residual may be a phase residual or an orthogonal component residual. The composite residual may be weighted by the reliability metric for the multi-dimensional symbol. The weighted value may be used to update the derotation phase once for the multi-dimensional symbol, or once for each of the individual symbols in the multi-dimensional symbol r_k^D.

Other systems, methods, features and advantages of the invention will be or will become apparent to one with skill in the art upon examination of the following figures and detailed description. It is intended that all such additional systems, methods, features and advantages be included within this description, be within the scope of the invention, and be protected by the accompanying claims.

BRIEF DESCRIPTION OF THE FIGURES

The invention can be better understood with reference to the following figures. The components in the figures are not necessarily to scale, emphasis instead being placed upon illustrating the principles of the invention. Moreover, in the figures, like reference numerals designate corresponding parts throughout the different views.

Figure 1 is a block diagram of a TCM encoding system based on a rate k/n encoder.

Figure 2 illustrates a rate 6/6 systematic convolutional encoder.

Figure 3 illustrates the two 8-PSK symbol constellations for each of the 8-PSK symbols output by the encoder of Figure 5.

Figure 4A is a block diagram of a system in accordance with the subject invention.

Figure 4B is a block diagram of an application of the system of Figure 4A.

Figures 5A-5C illustrate examples of the computation of a weighted phase residual.

Figures 6A-6C illustrate examples of the computation of a weighted orthogonal component residual.

Figure 7 is a flowchart of a method of computing reliability metrics for a multi-dimensional channel symbol.

Figure 8 is a portion of a trellis diagram illustrating an example of recursive calculation of forward probabilities.

Figure 9 is a portion of a trellis diagram illustrating an example of recursive calculation of backward probabilities.

Figure 10 is a portion of a trellis diagram illustrating an example of calculation of joint probabilities from the forward, backward, and edge probabilities.

Figure 11 is a portion of an example trellis diagram illustrating, for a two-state, symmetric trellis, computation of log-likelihood values.

DETAILED DESCRIPTION

A. Multi-Dimensional Channel Symbols

Multi-dimensional channel symbols are channel symbols which have multiple dimensions that bear a relationship to one another. Typically, this relationship derives from the fact that the multiple dimensions are encoded from the same underlying source bits. Multi-dimensional channel symbols may be produced by a trellis-coded modulation (TCM) encoding system. The TCM encoding system may in turn incorporate a rate k/n convolutional encoder.

Figure 1 illustrates a TCM encoding system which incorporates a rate k/n encoder. In general, the integer k is less than or equal to the integer n. The ratio of k to n relates to the redundancy of the symbols coded by the encoder. The greater this ratio of source bits to coded bits, the less redundancy is built into the encoded symbols.

As illustrated, the TCM encoding system comprises a serial to parallel (S/P) converter 402, a rate k/n encoder 406, a bit to symbol mapper 410, and, optionally, a symbol multiplexor 414. Incoming source bits 400 are serially input to S/P converter 402. S/P converter 402 converts the serial stream of input bits to successive parallel renditions of k bits each. Each k bit rendition 404 is input to a rate k/n encoder 406. The output of the rate k/n encoder comprises successive parallel renditions of n bits each. Each n bit rendition 408 is input to bit to symbol mapper 410. Bit to symbol mapper 410 converts each rendition 408 of n bits to a D-dimensional channel symbol, where D is an integer equal to 1 or more. In the case in which D=1, the symbol multiplexor 414 is unnecessary. In the case in which D>1, the symbol multiplexor 414 serializes the D components of a D-dimensional symbol and outputs the same on signal line 416. In one implementation, the multiplexor serializes the D components two at a time to represent the I and Q components of a quadrature output.

Figure 2 illustrates an example of a rate k/n encoder which can be used in the TCM encoding system of Figure 1. The particular example shown is a rate 6/6 systematic, convolutional encoder. As illustrated, the input to the encoder is a 6-tuple of input bits which can be represented by (u_5, u_4, u_3, u_2, u_1, u_0), and the output of the encoder is an output symbol comprising a 6-tuple of output bits which can be represented as (y_2^1, y_1^1, y_0^1, y_2^0, y_1^0, y_0^0).

The 6 output bits of the encoder can be grouped into two 3-tuples, represented respectively as (y_2^1, y_1^1, y_0^1) and as (y_2^0, y_1^0, y_0^0), and the mapper 410 maps each such 3-tuple into an 8-PSK symbol. Each of the two 8-PSK symbols is a quadrature symbol having I and Q components. A 6-tuple output symbol can be considered to be a single four-dimensional symbol, since each output symbol is comprised of two 8-PSK symbols, and each 8-PSK symbol is represented in two dimensions on the (I and Q) complex plane. Figure 3 illustrates the functioning of the bit to symbol mapper 410 in this example. The particular mapping which is used can be represented by the following table:

3-tuple (y_2^i, y_1^i, y_0^i), i=0,1	8-PSK symbol
(0, 0, 0)	$\pi/16$
(0, 0, 1)	$3\pi/16$
(0, 1, 1)	$5\pi/16$
(0, 1, 0)	$7\pi/16$
(1, 1, 0)	$9\pi/16$
(1, 1, 1)	$11\pi/16$
(1, 0, 1)	$13\pi/16$
(1, 0, 0)	$15\pi/16$

In this particular example, a Gray mapping is employed, in which adjacent symbols correspond to 3-tuples which differ by no more than a single bit. An alternate Gray mapping is also possible with 8-PSK since, as is known, there are two unique Gray maps for 8-PSK. It should be appreciated, however, that, in the general case involving a rate k/n (or n/n) encoder, Gray mapping need not be employed.

This example can easily be generalized to other cases. Consider, for example, a rate 4/4 encoder in which the four bits output by the encoder are grouped into two 2-bit tuples, each of which is mapped into a QPSK symbol. Again, an output symbol can be considered to be a single 4-dimensional symbol, since two QPSK symbols are output, and each symbol is two-dimensional.

Next, consider a rate 12/12 encoder in which the 12 bits output by the encoder are grouped into 3 4-bit tuples, each of which is mapped into a 16-QAM symbol. An output symbol can be considered to be a 6-dimensional symbol, since each output symbol is comprised of three 16-QAM symbols, and each 16-QAM symbol is two-dimensional, i.e., is represented by I and Q coordinates.

Note that, in general, a multidimensional code is a rate k/n code where $k \leq n$, and n is evenly divisible by the number of constellation points within a modulation symbol (e.g., 4 constellation points for a single QPSK-modulated symbol). Here, examples involving rate n/n codes have been used simply for ease of illustration. However, examples of the more general k/n case can easily be extrapolated from this disclosure.

B. Embodiments of the Invention

A first embodiment of a system 100 in accordance with the invention is illustrated in Figure 4A. As illustrated, the system 100 comprises a symbol estimation module 102 coupled to a residual determination module 104. A multi-dimensional symbol is input to the symbol estimation module 102. The multi-dimensional symbol can be represented by the vector r_k^D, where the subscript k indicates time, and the superscript D indicates that the vector comprises a plurality D of individual

constituent symbols. The symbol estimation module 102 provides an estimate for the multi-dimensional channel symbol. The estimate itself may be multi-dimensional, and represented by the vector s_k^D. This indicates that estimates of the individual constituent symbols are included within or easily derivable from s_k^D.

The residual determination module 104 forms a multi-dimensional residual indicating a difference between the symbol r_k^D and the estimate s_k^D. The residual may be a scalar z_k and lack multiple dimensions, or it may be a vector z_k^D, and have multiple dimensions, each representing a residual between an individual constituent symbol, and the estimate of that individual symbol. A function of the vector z_k^D, which can be designated $f(z_k^D)$, may also be provided by module 104. The residuals may be, without limitation, phase residuals or they may be orthogonal component residuals, i.e., the component of a symbol orthogonal to the estimate of that symbol.

A reliability metric R_k, indicating the reliability of the multi-dimensional estimate, may also be provided by the symbol estimation module 102. The reliability metric may be used by the residual determination module 104 to weight the corresponding residuals. If the residual is the vector z_k^D, the weighted residual may be represented by $R_k \bullet z_k^D$, indicating that R_k is used to weight each of the components of z_k^D. If the residual is the scalar z_k, the weighted residual may be represented by $R_k \bullet z_k$. If a function $f(z_k^D)$ is computed, the weighted value $R_k \bullet f(z_k^D)$ may be provided by module 104. Module 104 may also form a joint function of the reliability metric R_k and the residual z_k^D, which is designated $g(R_k, z_k^D)$ in the figure. One of ordinary skill in the art will appreciate from a reading of this disclosure that many more example are possible.

In one application, illustrated in Figure 4B, a phase determination module 106 is provided which determines, responsive to one or more of the weighted or unweighted residuals from module 104, a derotation phase for the multi-dimensional symbol. This derotation phase may be a scalar θ_k, which applies to each of the

individual symbols in the multi-dimensional symbol. Or it may be a multi-dimensional vector θ_k^D, including individual derotation phases for each of the individual constituent symbols in the multi-dimensional symbol. These derotation phases may be used to derotate the individual symbols in the multi-dimensional symbol to reduce or eliminate phase offset with respect to the carrier.

In this application, the residual determination module and the phase determination module 106 may together form a tracking loop module 108. The derotation phase output from the tracking loop module 108 may be used to derotate the individual symbols in the multi-dimensional symbol r_k^D as described in any of the embodiments employing feedfoward tracking loops disclosed in Howrey Dkt. No. 01827.0042.US00, Conexant Dkt. No. 00CXT0361D, or Howrey Dkt. No. 01827.0043.US00, Conexant Dkt. No. 00CXT0362D, both previously incorporated herein by reference.

In another application, also illustrated in Figure 4B, a phase determination module 106 is provided which determines, responsive to the weighted or unweighted residuals, a phase offset estimate for the multi-dimensional symbol. The phase offset estimate may be a scalar $\Delta\theta_k$, which applies to each of the individual symbols of the multi-dimensional symbol r_k^D. Or it may be a vector $\Delta\theta_k^D$, with individual phase offset estimates for each of the individual constituent symbols in the multi-dimensional symbol. These phase offset estimates may be used to derotate the individual symbols in the multi-dimensional symbol to reduce or eliminate phase offset with respect to the carrier.

In this application, the residual determination module and the phase determination module 106 may together form a tracking loop module 108. The phase offset estimates output from the tracking loop module 108 may be used by an accumulator to form derotation phases which in turn are used to derotate the individual symbols in the multi-dimensional symbol r_k^D as described in any of the

embodiments employing feedback tracking loops disclosed in Howrey Dkt. No. 01827.0042.US00, Conexant Dkt. No. 00CXT0361D, or Howrey Dkt. No. 01827.0043.US00, Conexant Dkt. No. 00CXT0362D, both previously incorporated herein by reference.

The module 102 may be any device capable of producing soft symbol estimates, including, without limitation, a MAP decoder, a log-MAP decoder, a Soft Input Soft Output ("SISO") decoder, a SOVA decoder, a max-log MAP decoder, or the like. Module 102 may also be a Viterbi decoder where a reliability metric is derived from the best path metric minus the second-best path metric. In one implementation, a modified log-MAP decoder is used where only the alpha (forward recursion) engines are utilized. Additional information about the log-MAP decoding process is available in U.S. Patent Application Serial No. Not Yet Assigned, Howrey Dkt. 01827.0041.US00, Conexant Dkt. No. 00CXT0360D, and U.S. Patent Application Serial No. Not Yet Assigned, Howrey Dkt. No. 01827.0042.US00, Conexant Dkt. No. 00CXT0361D, both of which were previously incorporated herein by reference.

In a second embodiment of the invention, the symbol estimation module is a maximum a posteriori (MAP) decoder which utilizes a MAP algorithm to determine an estimate of a multi-dimensional channel symbol, and, optionally, a reliability metric for the estimate. For additional details on MAP decoders, the reader is referred to "Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate," L.R. Bahl et al., IEEE Transactions on Information Theory, March 1974, pp. 27-30 (hereinafter referred to as "the Bahl reference"); "Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes," C. Berrou et al., Proc. ICC '93 Geneva, Switzerland, May 1993, pp. 1064-1070 (hereinafter referred to as "the Berrou reference"); "An Intuitive Justification and a Simplified Implementation of the MAP Decoder for Convolutional Codes," A. Viterbi, IEEE Journal On Selected Areas In Telecommunications, Vol. 16, No. 2, Feb. 1998, pp. 260-264 (hereinafter referred

to as "the Viterbi reference"); and J. Hagenauer and P. Hoeher, "A Viterbi Algorithm with Soft-decision outputs and its applications," in Proceedings of IEEE Globecom '89, Dallas, TX, Nov. 1989, pp. 47.1.1-47.1.7. ("the Hagenauer reference") Each of the Bahl, Berrou, Viterbi, and Hagenauer references are hereby fully incorporated by reference herein as though set forth in full.

The MAP decoder may be a log-MAP decoder in which the underlying probabilities are expressed in the natural-log domain. Log domain probabilities are advantageous since they ease computational burden. For example, with log domain probabilities, joint probabilities can be computed through addition rather than multiplication. It should be appreciated, however, that the process can easily be generalized to the case in which the probabilities are expressed in terms of their normal domains.

A flowchart of one embodiment of the process employed by this log-MAP decoder is illustrated in Figure 7. In this flowchart, the notation $\alpha_k(m)$ refers to the natural log domain (hereinafter referred to as "log domain") forward probability of being in state m at time k; γ_k^i refers to the log domain edge probability for edge i at time k; and $\beta_k(m)$ refers to the log domain reverse probability of being in state m at time k. It is assumed that a block of N multi-dimensional channel symbols is processed at a time.

In step 700, the boundary values $\alpha_0(m)$ and $\beta_N(m)$ are initialized for all values of m.

In step 702, for each observation r_k, $\alpha_k(m)$ and γ_k^i are computed for all values of m and i. Note that the "observation" r_k is a multi-dimensional channel symbol as perturbed by noise through passage through the channel. Advantageously, the forward probabilities $\alpha_k(m)$ are computed recursively as a function of $\alpha_{k-1}(m)$. In one implementation, this step may be performed using equations (21) and (23) from the Berrou reference.

An example of the computation of forward probabilities is illustrated in Figure 8. In this example, there are two permissible branches into state s_3, one from state s_1 and one from state s_2. The log domain probability of s_1, $\alpha(1)$, and the log domain probability of s_2, $\alpha(2)$, have been previously determined. Similarly, the log domain edge probabilities γ^1 and γ^2 have been previously determined. The objective is to compute the log domain probability of the state s_3, $\alpha(3)$, from the foregoing parameters.

In this example, the calculation of the probability $\alpha(3)$ can be expressed using the following equation:

$$\alpha(3) = MAX^*\left(\alpha(1) + \gamma^1, \alpha(2) + \gamma^2\right)$$

where the function $MAX^*(A,B)$ is defined to be:

$$MAX^*(A,B) \equiv MAX(A,B) + \ln\left(1 + \exp\left(-|A - B|\right)\right)$$

Turning back to Figure 7, in step 704, the reverse probabilities $\beta_k(m)$ are computed for all values of m. The probabilities $\beta_k(m)$ are advantageously computed recursively as a function of $\beta_{k+1}(m)$. In one implementation, this step may be performed using equation (22) from the Berrou reference.

An example of the computation of reverse probabilities is illustrated in Figure 9. In this example, there are two permissible branches into state s_6, one from state s_4 and one from state s_5. The log domain probability of s_4, $\beta(4)$, and the log domain probability of s_5, $\beta(5)$, have been previously determined. Similarly, the log domain edge probabilities γ^4 and γ^5, have been previously determined. The objective is to compute the log domain probability of the state s_6, $\beta(6)$, from the foregoing parameters.

In this example, the calculation of the natural log domain probability $\beta(6)$ can be expressed using the following equation:

$$\beta(6) = MAX^*\left(\beta(4) + \gamma^4, \beta(5) + \gamma^5\right)$$

where the function $MAX^*(A,B)$ is defined as before.

Turning back to Figure 7, in step 706, at the point where the forward and reverse probabilities are about to overlap, i.e., a point of adjacency, the joint log domain probabilities $\lambda_k^i = \alpha_k(m) + \gamma_k^i + \beta_{k+1}(m')$ are computed for all edges at the point of adjacency. An example of this process is illustrated in Figure 10. Referring to the leftmost state as state m at time k, and the rightmost state as state m' at time k+1, it is assumed that the forward state log domain probability $\alpha_k(m)$, the reverse state log domain probability $\beta_{k+1}(m')$, and the edge log domain probability γ_k^i have all been computed. This step involves adding these probabilities, i.e., performing the computation $\lambda_k^i = \alpha_k(m) + \gamma_k^i + \beta_{k+1}(m')$ in order to compute the joint log domain probability of transitioning between the two states along the prescribed edge.

Turning back to Figure 7, step 706 is followed by step 708. In step 708, a log-likelihood, LL_k, is determined at the point of adjacency for each of the possible symbols. The log-likelihood for a symbol S at time k, $LL_k(S)$, may be expressed by the following equation:

$$LL_k(S) = MAX^*_{\forall i \text{ that imply release of symbol } S}(\lambda_k^i) - MAX^*_{\forall i}(\lambda_k^i) \qquad (1)$$

This step can be explained with reference to Figure 11, which illustrates a portion of a trellis diagram for a log-MAP decoder configured in accordance with the invention. The particular example shown is a trellis diagram for a decoder configured for use in conjunction with the rate 6/6 encoder of Figure 5. Since there is only one storage element in the encoder, there are only two possible states of the encoder at a point in time, 0 or 1. Note that, in general, the state transitions are not symmetric as illustrated. Instead, in the general case for non-systematic codes involving more than two states, the trellis diagram exhibits a more elaborate "butterfly" pattern.

The leftmost portion of Figure 11 represents time k, and the rightmost portion represents time k+1. The two possible states at time k are indicated as state 0 and state 1, respectively, as are the two possible states at time k+1. It is assumed that the

portion of the trellis illustrated in Figure 11 is at the point of adjacency in the calculation of forward and reverse state probabilities. Thus, for time k, the forward state probabilities $\alpha_k(0)$ and $\alpha_k(1)$ have been computed for states 0 and 1, respectively. Similarly, for time k+1, the backward state probabilities $\beta_{k+1}(0)$ and $\beta_{k+1}(1)$ have been computed for states 0 and 1, respectively.

In addition, the edge probabilities γ_k^i have all been computed, as have the joint probabilities λ_k^i . In an encoder with k inputs and m storage elements, there will be $2^k \bullet 2^m$ possible edge transitions at a given point in time. For the encoder of Figure 5, consistent with this formula, there will be 128 possible edge transitions for a given point in time. This is reflected in the trellis diagram of Figure 11, in which 128 edge transitions are indicated, 64 emanating from state 0 at time k, and 64 emanating from state 1 at time k. These edges are all assigned a unique index ranging from 0 to 127. The corresponding edge probabilities are referred to as γ_k^i, where i ranges from 0 to 127. Similarly, the corresponding joint probabilities are referred to as λ_k^i, where i ranges from 0 to 127.

Each of the edges is consistent with the release of a channel symbol. For an encoder with an n-tuple output, there are 2^n possible values of channel symbols. Consistent with this, in the encoder of Figure 5, there are 64 possible channel symbols. If the number of possible edge transitions at a point in time exceeds the number of possible channel symbols, there will be some edges which are consistent with the release of the same channel symbol. For the encoder of Figure 5, there will be two transitions which are consistent with the release of the same channel symbol. The possibility that more than one edge may imply release of the same symbol is reflected in the first MAX^* term equation (1) above.

Turning back to Figure 7, step 708 is followed by step 710, in which the channel symbol at time k is estimated, along with the reliability metric for that symbol, responsive to the LL_k values computed in the previous step. In one

embodiment, the estimated symbol is taken to be the symbol p which corresponds to the maximum value of LL_k. This condition may be represented by the following equation:

$$LL_k(p) = MAX_{\forall \text{ possible channel symbols } s}(LL_k(S))$$

Step 710 is followed by step 712, when a reliability metric for the estimate p is derived from $LL_k(p)$. The reliability metric for p may be equal to $LL_k(p)$. Alternatively, that metric may be set equal to $\exp(LL_k(p))$. In general, other examples are possible where the reliability metric is a function of LL_k.

The foregoing process may then be repeated at other points of adjacency in the trellis diagram and for additional blocks of incoming symbols. In one embodiment, the process iterates until each of the symbols in a block is estimated, and reliability metrics for each of these estimates determined. Then, the process may be repeated for additional blocks.

In a third embodiment of the invention, the engine 104 is a modified log-MAP decoder in which only the alpha (forward recursion) engines function. These alpha engines compute the forward recursive probabilities $\alpha_k(m)$ and the edge probabilities γ_k^i. Then, they compute the joint probabilties λ_k^i in accordance with the following equation: $\lambda_k^i = \alpha_k(m) + \gamma_k^i$. The log-likelihoods, LL_k, are then computed from the joint probabilities λ_k^i as described in the previous embodiment. The reliability metrics for the multi-dimensional symbols are then computed as described in the previous embodiment.

A computer readable medium which tangibly embodies the method steps of any of the foregoing embodiments is within the scope of the invention. Such a medium may include, without limitation, RAM, ROM, EPROM, EEPROM, floppy disk, hard disk, CD-ROM, etc. The invention also includes the method steps of any of the foregoing embodiments synthesized as digital logic in an integrated circuit, such as an FPGA, or PLA, for example.

Several advantages flow from the production of a single estimate for all dimensions of a multi-dimensional channel symbol rather than separate estimates for the different dimensions of the symbol. First, the approach is more efficient from a computational standpoint. Second, it is more accurate since, in effect, it uses distance statistics in multi-dimensional space to determine an estimate of a multi-dimensional symbol. In contrast, in an approach where an estimate is separately produced for each constituent symbol, the distances used amount to projections of true distances onto a plane. In some instances, this could result in a less accurate estimate, or diminished tracking performance by the tracking loop.

C. Examples

In one example of the operation of residual determination module 104, the residual which is produced is the vector z_k^D, and the same reliability metric R_k is used to weight each of the components of the vector z_k^D. The tracking loop module in this example determines an individual component of θ_k^D or $\Delta\theta_k^D$, as the case may be, responsive to the corresponding individual weighted residual. In other words, it determines θ_k^i or $\Delta\theta_k^i$, $1 \leq i \leq D$, responsive to $R_k \bullet z_k^i$, $1 \leq i \leq D$.

In a second example, a function of the individual component residuals of z_k^D is formed. Such a function may be designated as $f(z_k^D)$. For example, $f(z_k^D)$ may be the average of the individual component residuals. This function may be weighted by the reliability metric for the multi-dimensional channel symbol. The tracking loop module may then determine a scalar derotation phase θ_k or phase offset estimate $\Delta\theta_k$ responsive to the value $R_k \bullet f(z_k^D)$. This value may then be applied to each of the individual symbols in the multi-dimensional symbol.

In a third example, a composite residual is formed for a multi-dimensional symbol. The composite residual may be a phase residual or an orthogonal component residual. The composite residual may be weighted by the reliability metric for the multi-dimensional symbol. The weighted value may be used to update the derotation

phase once for the multi-dimensional symbol, or once for each of the individual symbols in the multi-dimensional symbol r_k^D.

Figures 5A-5C and 6A-6C illustrate several additional examples of the operation of residual determination module 104. In Figure 5A, a phase residual Φ_{k-1} between the individual symbol r_{k-1} and as estimate of that symbol s_{k-1} is determined in accordance with the following expression:

$$\Phi_{k-1} = angle(r_{k-1} s*_{k-1})$$

where * is the complex conjugate relation. This phase residual may be weighted by the reliability metric R_k for the multi-dimensional symbol of which the individual symbol r_{k-1} is a part in order to determine o_{k-1}, the input to the phase determination module 106.

In Figure 5B, a phase residual Φ_k between the individual symbol r_k and the estimate of that symbol s_k is determined in accordance with the following expression:

$$\Phi_k = angle(r_k s_k *)$$

where * is the complex conjugate operation. This phase residual may be weighted by the reliability metric R_k for the multi-dimensional symbol of which the individual symbol r_k is a part in order to determine o_k, an input to the phase determination module.

Figure 5C illustrates another approach in which the composite phase residual Φ_k for both individual symbols r_{k-1} and r_k may be determined in accordance with the following expression:

$$\Phi_k = angle(r_k s*_k + r_{k-1} s*_{k-1})$$

This composite phase residual Φ_k may then be weighted by the reliability metric R_k for the multi-dimensional symbol comprising the individual symbols r_{k-1} and r_k to

form the input o_k to the phase determination module. This value may be input to the phase determination module once for the multi-dimensional symbol comprising r_{k-1} and r_k, or twice, once for r_{k-1} and once for r_k.

Note that, since the underlying phase intended to be tracked will generally not vary much from symbol to symbol, the composite phase residual of Figure 5C will typically be about equal to either of the phase residuals computed in Figures 5A and 5B – when no additive noise (such as AWGN) is present.

In Figure 6A, the orthogonal component residual e_{k-1}, the component of the individual symbol r_{k-1} which is orthogonal to the estimated symbol s_{k-1}, is formed in accordance with the following expression:

$$e_{k-1} = imag(r_{k-1} s^*_{k-1})$$

where * is the complex conjugate operation. This component residual may then be weighted by the reliability metric R_k for the multi-dimensional symbol of which the individual symbol r_k is a part to produce o_{k-1}, the input to the phase determination module.

In Figure 6B, the orthogonal component residual e_k, the component of the symbol r_k which is orthogonal to the estimated symbol s_k, is formed in accordance with the following expression:

$$e_k = imag(r_k s_k^*)$$

where * is the complex conjugate operation. This component residual may then be weighted by the reliability metric R_k for the multi-dimensional symbol of which the individual symbol r_k is a part to form o_k, the input to the parameter determination module.

Figure 6C illustrates another approach in which a composite orthogonal component residual e_k is formed in accordance with the following expression:

$$e_k = imag(r_k s^*_k + r_{k-1} s^*_{k-1})$$

where * is the complex conjugate operation.

This composite orthogonal component residual may then be weighted by the reliability metric R_k for the multi-dimensional symbol to form the input o_k to the parameter determination module. This value may be input to the phase determination module once for the multi-dimensional symbol comprising r_{k-1} and r_k, or twice, once for r_{k-1} and once for r_k.

Note that, if the phase noise process (carrier phase variation) is slow, the orthogonal component of individual symbols will generally not vary much from symbol to symbol, and so the composite orthogonal component residual of Figure 6C will typically be about equal to the orthogonal components residuals of Figures 6A and 6B.

Examples are also possible in which statistics derived from vector operations on symbols may be used as inputs to the parameter determination module. Vectors of received symbols $\mathbf{r} = [r_k\ r_{k-1}\ \ldots r_N]$ and estimated symbols $\mathbf{s} = [s_k\ s_{k-1}\ \ldots s_N]$ may be assembled. Then, the statistic $c = \mathbf{r}\ \mathbf{s}^H$, where 'H' indicates the Hermetian (conjugate transpose operation), is performed. The result is a scalar, and can be expressed in scalar terms by $c = r_k\ s^*_k + r_{k-1}\ s^*_{k-1} + \ldots\ r_N\ s^*_N$. The phase of c may then be computed. This phase may then optionally be weighted by the reliability of the multi-dimensional symbol estimate, and the weighted value input to the phase determination module at the arrival/reception rate of multi-dimensional symbols. Alternately, this weighted value may be input to the phase determination module at the individual symbol arrival rate.

Also, in lieu of the phase of c, the imaginary part of c may be used to compute the input to the tracking loop. In particular, this imaginary part may be weighted by the multi-dimensional symbol reliability, and then input to the phase determination module at either the multi-dimensional or individual symbol arrival rate.

While various embodiments of the invention have been described, it will be apparent to those of ordinary skill in the art that many more embodiments and implementations are possible that are within the scope of this invention. Accordingly,

the invention is not to be restricted except in light of the attached claims and their equivalents.